Grammar

Last modified: 2026-11-13 12:24:54 (PST)

> **AI-Assisted Authorship**
>
> This chapter was written with the assistance of GitHub Copilot, which expanded on outlined ideas and draft notes provided by the author. The content represents the author's perspective and has been reviewed for accuracy, but the detailed prose was generated through AI assistance.

Clear grammar removes friction between your ideas and your reader. This chapter collects a few grammatical habits that consistently make scientific writing easier to read.

Relative pronouns

Relative pronouns introduce a clause that describes a preceding noun. The relative pronouns are:

- *that*
- *which*
- *who*
- *whom*
- *whose*

Two habits make these clauses easier to read: choosing between *that* and *which* correctly, and keeping the relative pronoun rather than dropping it.

Restrictive vs. non-restrictive clauses: *that* vs. *which*

Definition 0.1 (Restrictive clause)**.** A **restrictive clause** is a relative clause that identifies which thing you mean, and that cannot be removed without changing the meaning of the sentence.

Definition 0.2 (Non-restrictive clause)**.** A **non-restrictive clause** is a relative clause that adds information the reader does not need in order to know which thing you mean.

Use *that* for a restrictive clause. Use *which*, set off by commas, for a non-restrictive clause.

Example 0.1 (Restrictive clause (*that*))**.**

> The dataset **that** we collected in 2023 contains 4,000 records.

The clause "that we collected in 2023" tells the reader which dataset we mean. Removing it changes the meaning, so the clause is restrictive and takes *that* with no comma.

Example 0.2 (Non-restrictive clause (*which*))**.**

> Our 2023 dataset, **which** contains 4,000 records, is publicly available.

The clause "which contains 4,000 records" adds a detail, but the reader already knows which dataset we mean. The clause is non-restrictive, so it takes *which* and is set off by commas.

A quick test: if you can drop the clause without changing which thing you are describing, use *which* and commas; otherwise, use *that*.

Do not omit relative pronouns

English lets you drop a relative pronoun in some sentences, but keeping it makes the structure of the sentence explicit and easier to parse on the first read. Prefer to keep *that*, *which*, or *who*, even when the grammar would let you drop it.

Example 0.3 (Restoring an omitted relative pronoun)**.**

> ✗ This body of work laid the methodological groundwork I now apply to enteric fever.
>
> ✓ This body of work laid the methodological groundwork **that** I now apply to enteric fever.

The word "that" signals immediately that a describing clause is beginning, so the reader does not briefly misread "the methodological groundwork I" as a single phrase and then have to backtrack.

Demonstrative pronouns

There are four demonstrative words:

- *this*
- *that*
- *these*
- *those*

They can serve two roles. As adjectives, they modify a noun: "**this** result", "**those** samples". As standalone pronouns, they replace a noun: "**This** shows…", "**That** means…".

When a demonstrative stands alone as a pronoun, the reader has to search backward for its referent — and often the referent is a whole preceding clause rather than a single noun, which leaves the reference ambiguous. Prefer to follow a demonstrative with the noun it points to, so the reference is explicit. The adverbs *here* and *there* point the same way: "getting there" leaves the reader to guess where "there" is.

A demonstrative at the end of a sentence has no room after it for a naming noun, so a bare sentence-final *this* or *that* always leaves its referent unnamed. End the sentence with the noun instead.

Example 0.1 (Naming the referent)**.**

✗ The model overfit the training data. **This** poses a problem for deployment.

✓ The model overfit the training data. **This overfitting** poses a problem for deployment.

✗ The first model's estimates were biased upward. The second model avoids **this**.

✓ The first model's estimates were biased upward. The second model avoids **this bias**.

Adding the nouns "overfitting" and "bias" states exactly what each "this" refers to, instead of leaving the reader to infer it. The second pair ends its sentence with the demonstrative, so the fix has to add the noun after it, at the very end.

Quantifiers used as pronouns

Words that say how many, or how much, of something cause the same problem as demonstratives when they stand alone in place of a noun. Grammars call these words *quantifiers*, and call a quantifier that stands alone an *indefinite pronoun*. The common ones are:

- *both*
- *all* (and counted forms such as *all three*)
- *each*
- *any*
- *some*
- *most*
- *many*
- *few*
- *none*

Every belongs to the same family, but it cannot stand alone ("every" needs a noun, as in "every model"), so it never causes this problem.

You do not need a complete list of quantifiers to apply the advice. Whenever a quantifier stands where a noun should be, the reader has to search backward to find out what it quantifies. Follow the quantifier with the noun: "**both models**", "**all three estimators**", "**each interval**". An "of" phrase that names the noun works too: "**none of the intervals**".

Example 0.1 (Naming what a quantifier refers to)**.**

> ✗ We fit a linear model and a spline model to the 2019 and 2020 surveys. **Both** return the same fit.
>
> ✓ We fit a linear model and a spline model to the 2019 and 2020 surveys. **Both models** return the same fit.

In the first version, "both" could mean the two models or the two surveys. Adding "models" settles which pair the sentence is about.

Avoid garden-path sentences

A *garden-path sentence* leads the reader into one reading and then forces them to go back and reread when the rest of the sentence does not fit (Wikipedia: Garden-path sentence). The sentence may be grammatical, but the reader parses it twice.

A common cause of garden-path sentences in technical writing is a clause that describes a noun but has lost its opening *that* or *that were*. Without the pronoun, the reader takes the

noun and the start of the clause as one phrase and has to reattach the words that follow. Do not omit relative pronouns gives the general rule.

When you reread a sentence and stumble, find the point where your first reading went wrong, and rewrite the sentence so that the first reading is the right one.

Example 0.1 (Rewriting a garden-path sentence)**.**

> ✗ Report only SHAs a command you ran printed in full.
>
> ✓ Report only SHAs that appeared in full in the output of a command that you ran.

In the first version, the reader takes "SHAs a command you ran" as one phrase, then reaches "printed" and has to work out that "a command you ran printed in full" is a clause describing the SHAs. The rewrite marks the clause with "that" and keeps each verb next to its own subject.

Name the object of a relational noun

Some nouns name a relation, so they are incomplete without the thing they relate to. Relational nouns include:

- *cause* (of what?)
- *effect* (of what, on what?)
- *reason* (for what?)
- *result* (of what?)
- *example* (of what?)

When one of these nouns appears without its "of" or "for" phrase, the reader has to recover the missing object from context, just as with a pronoun that has no clear referent. Write the object next to the noun.

Example 0.1 (Comparing four versions of one sentence)**.** Each version tries to say what usually causes garden-path sentences.

> 1. A common cause in technical writing is a clause that describes a noun but has lost its opening *that* or *that were*.
> 2. In technical writing, a common cause is a clause that describes a noun but has lost its opening *that* or *that were*.
> 3. In technical writing, a common cause of garden-path parsing is a clause that describes a noun but has lost its opening *that* or *that were*.

> 4. A common cause of garden-path sentences in technical writing is a clause that describes a noun but has lost its opening *that* or *that were*.
>
> Versions 1 and 2 never say what the clause causes, so the reader has to look back to find what "cause" refers to. Version 3 names the object, "garden-path parsing", but its fronted phrase delays the subject (see Lead with the subject), and "garden-path parsing" introduces a second term for the "garden-path sentences" that the section defines. Version 4 names the object, uses the defined term, and starts with the subject.

Appositive fronting

An *appositive* is a noun phrase that renames or describes the noun beside it. In "Bovine respiratory disease (BRD), a disease complex with a multifactorial etiology, kills calves", "a disease complex with a multifactorial etiology" is an appositive of "BRD". *Appositive fronting* moves the appositive to the start of the sentence, before the noun it describes, which then follows as the subject: "A disease complex with a multifactorial etiology, BRD is a leading cause of mortality in calves." Often the subject becomes a pronoun, as in "A disease complex with a multifactorial etiology, it is a leading cause of mortality in calves", which needs an antecedent for "it" in the sentence before.

Fronting is a stylistic choice, with costs and benefits.

Benefits

- It states the category before the specifics. The reader learns what kind of thing the subject is, then what is said about it.
- It joins two statements without a connecting verb such as "is" or "serves as", so the sentence is shorter.
- It puts a familiar description first and the new claim last, where the reader's attention lands.

Costs

- The fronted phrase must describe the grammatical subject that follows. Otherwise it dangles: "A disease complex with a multifactorial etiology, mortality in calves is high" says that mortality is a disease complex.
- The reader waits for the subject and verb, which runs against the advice to lead with the subject. A long fronted appositive delays them, and a reader may take the phrase for the subject (compare Avoid garden-path sentences).
- The sentence no longer asserts the description as a main-clause claim, so the reader may treat it as background even when it is the point.

- A pronoun subject ("it") needs an antecedent in the previous sentence. A reader who cannot find it must stop to look.
- It seems more formal and literary, which suits some venues and not others.

Guideline. Front an appositive when it is short, familiar to the reader, and describes the subject. Keep the plain "X is Y" form when the description is the new information the paper depends on, when it is long, or when the subject is a pronoun without a clear antecedent.

Example 0.1 (Plain and fronted forms)**.**

> Plain: Bovine respiratory disease (BRD) is a disease complex with a multifactorial etiology and serves as a significant cause of mortality among preweaned calves and weaned cattle alike.
>
> Fronted: A disease complex with a multifactorial etiology, BRD is a leading cause of mortality in both preweaned calves and weaned cattle.

The fronted version is shorter and puts the category first. It works here because "a disease complex with a multifactorial etiology" describes BRD, the subject that follows it, and the subject is named, not a pronoun. If the multifactorial etiology is what the paper is about, the plain form states it as a claim the reader can challenge, and is the better choice.

Put a comma before the last item of a list

In a list of three or more items, put a comma before the "and" or "or" that closes the list. This comma is called the *serial comma*, or the *Oxford comma*. With it, the reader sees where each item ends, and does not have to guess whether the last two items form a pair.

Write "the mean, the median, and the mode", not "the mean, the median and the mode". The same holds when the items are long, are split across source lines, or contain math.

Example 0.1 (Adding the serial comma)**.**

> ✗ We compare the lasso, ridge regression and elastic net.
>
> ✓ We compare the lasso, ridge regression, and elastic net.

In the first version, a reader can take "ridge regression and elastic net" as a single method that combines the two. The comma before "and" shows that there are three methods.

A list of two items takes no comma: "the mean and the median". A phrase that opens a sentence is set off by its own comma, and it does not make a list: "In practice, the mean and the median agree."

A line that must keep a list without the serial comma, such as a quotation or the ✗ half of an example, takes the marker `<!-- check-serial-comma: allow -->`.